WE CLAIM

1. A data processor core comprising:

a clock signal input operable to receive a processor clock signal;

a memory access interface portion operable to perform data transfer operations between an external data source and at least one memory associated with said data processor core in response to receipt of said processor clock signal;

a data processing portion operable to perform further data processing operations in response to receipt of said processor clock signal;

at least one further input operable to receive a memory access enable signal;

at least one read/write port operable to receive and send data via a bus to said at least one memory associated with said data processor core; wherein

said memory access interface portion is operable to receive said processor clock signal when said memory access enable signal has a predetermined value and not to receive said processor clock signal when said memory access enable signal does not have said predetermined value; and

said data processing portion is operable to receive said processor clock signal when a data processing enable signal has a further predetermined value and not to receive said processor clock signal when said data processing enable signal does not have said further predetermined value.

2. A data processor core according to claim 1, said data processor core further comprising data processing enable logic operable to generate said data processing enable signal.

3. A data processor core according to claim 2, wherein said data processing enable signal is operable to generate said data processing enable signal in response to detection of a state of said processor clock and a state of said bus in data communication with said read/write port.

4. A data processor core according to claim 1, comprising a second further input, said second further input being operable to receive said data processing enable signal.

5. A data processor core according to claim 1, wherein said predetermined value and said further predetermined value are the same.

6. A data processor core according to claim 1, wherein said memory access interface is operable to transfer data to or from said at least one memory via said read/write port and said bus.

7. A data processor core according to claim 1, wherein said external data source comprises a further memory associated with said processor core.

8. A data processor core according to claim 1, wherein said external data source comprises a further memory, said memory access interface being operable to transfer data to and from said further memory via a direct memory access controller.

9. A data processor core according to claim 1, wherein said further memory comprises a flash memory operable to store boot up code and said at least one memory comprises an instruction memory, said memory access interface portion being operable to transfer said boot up code from said flash memory to said instruction memory in response to receipt of said processor clock.

10. A data processor core according to claim 1, wherein said memory access enable signal comprises a clock signal having a different frequency to said processor clock signal and periodically, at said different frequency, going from a first state having said predetermined value to a second state not having said predetermined value, said memory interface being operable to receive said processor clock in response to said memory enable signal being in said first state and not to receive said processor clock in response to said memory enable clock signal being in said second state.

11. A data processor core according to claim 1, further comprising at least one logic gate operable to combine said processor clock signal received at said processor clock signal input with said processor clock enable signal, said at least one logic gate being operable to output said processor clock signal when said processor enable signal has said further predetermined value and not to output said processor clock signal when said processor enable signal does not have said further predetermined value, said data processing portion receiving said output of said at least one logic gate such that said data processing operations performed by said data processing portion are clocked by said output of said at least one logic gate.

12. A data processor core according to claim 1, further comprising at least one logic gate operable to combine said processor clock signal received at said processor clock signal input with said memory access enable signal received at said at least one further input, said at least one logic gate being operable to output said processor clock signal when said memory access enable signal has said predetermined value and not to output said processor clock signal when said memory access enable signal does not have said predetermined value, said memory access interface portion receiving said output of said at least one logic gate such that said data transfer operations performed by said memory access interface are clocked by said output of said at least one logic gate.

13. A data processor according to claim 1, said core further comprising:

arbitration logic associated with said read/write port; wherein

said arbitration logic is operable to route a data access request requesting access of data in one portion of said at least one memory received from said memory access interface to one of said at least two buses providing access to said one portion of said at least one memory and to route a further data access request requesting access of data in a further portion of said at least one memory received from said data processing portion to a further one of said at least two buses providing access to said further portion of said at least one memory, said routing of said data access requests being performed during the same clock cycle.

14. A data processor core according to claim 13, said arbitration logic being operable to select one of said at least two buses to route said data access request to, in

dependence upon an address location within said at least one memory associated with said data access request.

15. A data processor core according to claim 14, wherein said at least two portions of said memory comprise an instruction portion operable to store instructions and at least one data portion operable to store data items, said arbitration logic being operable to route said data access request to one of said at least two buses providing access to said instruction portion when data to be transferred is an instruction and to route said data access request to another of said at least two buses providing access to said at least one data portion when data to be transferred is a data item.

16. A data processor core according to claim 15, wherein said at least one data portion comprises two data portions, an even data portion operable to store data having an even address and an odd data portion operable to store data having an odd address, said read/write port being operable to transfer data between said processor core and said at least one memory via three buses, a first bus providing access to said instruction portion, a second bus providing access to said odd data portion and a third bus providing access to said even data portion, and said arbitration logic being operable to route a data access request to said first bus when data to be transferred is an instruction, to said second bus when data to be transferred is a data item associated with an odd address and to said third bus when data to be transferred is a data item associated with an even address.

17. A data processor core according to claim 13, wherein said arbitration logic is operable in response to receipt of a data access request from said memory access interface portion and a data access request from said data processing portion, both data access requests requesting access to data in one portion of said at least one memory, to route said data access request from said memory access interface portion to one of said at least two buses providing data access to said one portion of said at least one memory before routing said request from said processing portion to said one of said at least two buses.

18. A data processor core according to claim 13, said arbitration logic being operable to detect a wait request from at least one busy portion of said at least one memory, said arbitration logic being operable to not route any data access requests to said busy portion until said wait request is no longer detected.

19. A data processing apparatus comprising a data processor core according to claim 1, said data processing apparatus further comprising said at least one memory.

20. A data processing apparatus according to claim 19, said data processing apparatus further comprising a direct memory access controller operable to control transfer of data from said external data source to said at least one memory via said memory access interface.

21. A data processing apparatus according to claim 19, said data processing apparatus further comprising a flash memory and an instruction memory.

22. A data processing apparatus according to claim 19, wherein:

said at least one memory is divided into at least two portions; and

said data processing apparatus further comprising at least two buses, each bus allowing data access to a respective portion of said at least two portions of said at least one memory.

23. A data processing apparatus according to claim 22, wherein said at least one memory is divided into three portions, an instruction portion operable to store instructions, and two data portions, an even data portion operable to store data having an even address and an odd data portion operable to store data having an odd address, said data processing apparatus comprising three buses said read/write port being operable to transfer data between said processor core and said at least one memory via said three buses, a first bus providing access to said instruction portion, a second bus providing access to said odd data portion and a third bus providing access to said even data portion.

24. A data processing apparatus according to claim 22, wherein said at least one memory is a tightly coupled memory.

25. A method of transferring data between an external data source and a memory associated with a data processor core, said data processor core comprising a memory access interface portion operable to perform data transfer operations between said external data source and said memory associated with said data processor core and a

data processing portion operable to perform data processing operations, said method comprising the steps of:

receiving a processor clock signal and a memory access enable signal at the core;

performing said data transfer operations through said memory access interface portion clocked by said processor clock signal when said memory access enable signal has a predetermined value, and not performing said data transfer operations when said memory access enable signal does not have said predetermined value; and

performing said data processing operations at said data processing portion clocked by said processor clock signal when a processing enable signal has a further predetermined value and not performing said data processing operations when said processing enable signal does not have said further predetermined value.

26. A method according to claim 25, said method comprising the further step of generating said data processing enable signal using data processing enable logic present on said data processor core

27. A method according to claim 25, wherein said data processing enable signal is generated in response to detection of a state of said processor clock and a state of said bus in data communication with said read/write port.

28. A method according to claim 25, comprising the further step of receiving said data processing enable signal.

29. A method according to claim 25, wherein said predetermined value and said further predetermined value are the same.

30. A method according to claim 25, wherein said external data source comprises a further memory associated with said processor core.

31. A method according to claim 25, wherein said external data source comprises a further memory, said step of performing said data transfer operations comprising transferring data to and from said further memory via a direct memory access controller.

32. A method according to claim 25, wherein said further memory comprises a flash memory operable to store bootup code and said at least one memory comprises an instruction memory, said step of performing said data transfer operations comprising transferring said bootup code from said flash memory to said instruction memory in response to receipt of said processor clock signal.

33. A method according to claim 25, wherein said memory access enable signal comprises a clock signal having a different frequency to said processor clock signal and periodically, at said different frequency, going from a first state having said predetermined value to a second state not having said predetermined value.

34. A method according to claim 25, comprising a further step of combining said processor clock signal received at said processor clock signal input with said processor clock enable signal using at least one logic gate, and outputting said processor clock

signal from said at least one logic gate when said processor enable signal has said further predetermined value and not outputting said processor clock signal when said processor enable signal does not have said further predetermined value, and receiving said output of said at least one logic gate at said data processing portion such that said data processing operations performed by said data processing portion are clocked by said output of said at least one logic gate.

35. A method according to claim 25, comprising a further step of combining said processor clock signal with said memory access enable signal using at least one logic gate, said at least one logic gate being operable to output said processor clock signal when said memory access enable signal has said predetermined value and not to output said processor clock signal when said memory access enable signal does not have said predetermined value, said memory access interface portion receiving said output of said at least one logic gate such that said data transfer operations performed by said memory access interface are clocked by said output of said at least one logic gate.

36. A method according to claim 25, comprising the further steps of:

in response to a data access request requesting access of data in one portion of said at least one memory received from said memory access interface portion and a data access request requesting access to data in a further portion of said at least one memory received from said data processing portion, routing said data access request to one of at least two buses, said one of said at least two buses providing access to said one portion of said at least one memory, and routing said data access request received from said data processing portion to a further of said at least two buses, said further

bus providing access to said further portion of said at least one memory, said routing of said data access requests being performed during the same clock cycle.